UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/679,796	10/06/2003	Mary Czerwinski	MS1-3950US	3699

22801 7590 01/25/2010

LEE & HAYES, PLLC
601 W. RIVERSIDE AVENUE
SUITE 1400
SPOKANE, WA 99201

EXAMINER	
AUGUSTINE, NICHOLAS	

ART UNIT	PAPER NUMBER
2179	

NOTIFICATION DATE	DELIVERY MODE
01/25/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

lhptoms@leehayes.com

Office Action Summary	**Application No.** 10/679,796	**Applicant(s)** CZERWINSKI ET AL.	
	Examiner NICHOLAS AUGUSTINE	**Art Unit** 2179	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 November 2009*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-12,15-18,48-59,61-63,84-87,93 and 94* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-12,15-18,48-59,61-63,84-87,93 and 94* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

A. This action is in response to the following communications: Amendment filed: 11/04/2009. This action is made **Final**.

B. Claims 1-12, 15-18, 48-59, 61-63, 84-87, 93 and 94 remain pending.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

3. Claims 1-12, 15-18, 48-59, 61-63, 84-87, 93 and 94 rejected under 35 U.S.C. 103(a) as being unpatentable over DragThing in view of Westerman, Larry Alam (US Pat. 6,404,443 B1), herein referred to as “Westerman”.

Summary Note:

Three separate references are used to construct a single reference of DragThing. DragThing is a software application created in June 1995 with added features to as late as April 2002 (version 4.3) as cited in the DT2 material. *Version 1.0-4.3*

I. A DragThing Review from Applelinks herein referred to as (**APPLE**) (http://www.applelinks.com/reviews/dragthing-v4.shtml)

II. The about information off of the DragThing website using web.archive.org (Web archive, specializes in archiving the Internet since 1995) (http://web.archive.org/web/20020524155927/http://www.dragthing.com/) Herein referred to as (**DT1**)

III. Version history of DragThing from version 1.0 – 4.3 located on the main website of DragThing (http://www.dragthing.com/english/history4.html) which each update is dated with specific features added to the software application. Herein referred to as (**DT2**). The breaks down of the sited versions are as follows:

VERSION: DATE:

4.3	4-11-2002
4.0.1	4-19-2001
1.6	5-1-1996
1.5	1-1-1996
1.01	6-1-1996

It would be inherent that versions (4.3,4.0.1,1.6,1.5,1.01) from earliest to latest included the added functionality as disclosed by James Thomson (inventor of DragThing). Wherein such that the functionality of 1.01 was included in version 1.5, and the functionality of version 1.5 was included in 1.6, it is inherently known in the art of software versions that the programs add and fix functionality. Since the above versions are all about one solely software program product offering the combined teachings found at http://www.dragthing.com as well as the use of web archive (http://web.archive.org/web/20020409150945/www.dragthing.com/english/history.html) yields the prior art date of 4-09-2002 (with the release date of 4.3 being marked as 4-11-2002 as indicated on the above link from web archive.

Since the above mention references solely disclose one software application 'DragThing', all three references combined constitutes as a single reference 102 of a single product offering.

DragThing is copyright 1994-2002 as mentioned in the cited material.

Supplemental evidence: (http://en.wikipedia.org/wiki/Sorting_algorithm)

used to explain definitions commonly found in the art.

Secondary Reference: Westerman is introduced to show an obvious variation to the already substantial list of options provided by DragThing. Westerman will show also a method concerned with solving the problem of organization of windows and task by use of grouping these content items in a designated area much like DragThing, further Westerman will show us how to preserve window layouts.

As for independent claim 1, DragThing teaches a *method for managing* the at least two software applications on a display of a computer system comprising instantiating a first software application on the computer system; instantiating an other than first software application on the computer system; rendering the first software application in a first graphical window on a first portion of the display and in a first graphic control on a second portion of the display as a result of the instantiation of the first software application (DT2 pg.5-6, 18); rendering the other than the first software application in another than the first graphical window on the first portion of the display and in an other than the first graphic control on the second portion of the display as a result of the instantiation of the other than first software application;;(DT1, page 2, par.2; wherein the one or more graphical windows is an icon or button as described that represents a current running process or a shortcut of any component on the computing device, the second portion of the screen being that of the dock and the first portion being that of the desktop (maximize and minimize states) also according to paragraph one this software application is directed towards Macintosh computers, thus presenting the evidence of a computer system which runs this application; DT2, page 2-7; wherein mentioned is the ability to have processes placed in the dock upon runtime and contextual menus associated with the dock and items within the docks, multiple docks possible if desired by user); and while the first software application and the other than the first software application are instantiated (DT2, pg 5, 18, 19 and 21) obtaining an indication to organize the first graphic control and the other than first graphic control in a group, wherein obtaining the indication includes rendering a set of guides representing inclusion or exclusion from one or more possible grouping (docks) in the second portion of the display (DT2, page 5-6; DT1, page 2, par.1; user mouse event "drag" indicates a drag and drop action for organization; APPLE, page 2 paragraph 2; how DragThing performs every functionality of the well known "Dock" in Apple Macintosh operating system 10; http://en.wikipedia.org/wiki/Dock_(computing)); grouping the first and the other than first graphic controls on the second portion of the display according to the obtained indication(DT2, page 17 paragraph 2 and page 21 par.4; wherein the user can organize the arrangement of groups and content by means of dragging); and rendering the grouping in a bordered group in the second portion of the display, the bordered group including the first and the other than first graphic controls and a graphic control group (contextual menus, and group function controls (DT2, pages 2-7, 17-19 and 21, the group control tile configured for simultaneous action on both the first instantiated software application and the second instantiated software application; *(In DT2 provides evidence that suggests that there is an indication of running processes (applications concurrently running on the system), these processes are dock items, that dock items can be located in a plurality of docks and that multiple* *dock items can be selected and dragged to new locations)* (DT2, page 21, par.4 and page 18, par.2; page 12 and 6, wherein the user can group controls together in layers with group controls called tabs which are customizable by arrangement, color, font, name etc. In addition the user is presented with sorting options which will sort the application/icons by predefined conditions set in the options menu as defined by factory or user. Common to what is known in the art in the field of computational sorting is an automatic arrangement of data element defined by the rules of the sorting command currently executed; support for this reasoning can be found here: http://en.wikipedia.org/wiki/Sorting_algorithm).

DragThing does not specifically in detail explain that the system is capable of "preserving a layout, as rendered on the first portion of the display, of each graphical window that corresponds to each graphic control in the group". However in the same field of endeavor **Westerman teaches** preserving a layout, as rendered on the first portion of the display, of each graphical window that corresponds to each graphic control in the group (col.7, lines 40-46 to put this citation in better perspective In col.5, lines 5-32, 41-53; Westerman explains the grouping of windows and their presentation to the user). *It would have been obvious* to one of ordinary skill in the art at the time of the invention to combine Westerman into DragThing, this is true because Westerman teaches of solving the problem of organizing windows in to groups with the method of dismissing and calling groups of windows using the interface provided (col.2, lines 25-63). The combination of Westerman into

DragThing yields the variant option of preserving the content layout for an interface which manages grouped content (i.e. windows).

As for dependent claim 2, DragThing teaches the method as recited in claim 1, wherein obtaining an indication to organize the first and the other than the first graphic controls includes obtaining a user manipulation of a selection device to drag and drop the first graphic control on the other than the first graphic control (DT2, page 21, par.4; user drags controls together to form groups (docks (DT1, page 2, par 3 "… as many docks…") and tabs (DT2, page 18, par.2).

As for dependent claim 3, DragThing teaches the method as recited in claim 2 further comprising displaying a set of guides indicating one or more possible groupings of graphic controls corresponding to a drag and drop on a selected guide (DT2, page 5, last paragraph and page 6, first paragraph; indicator, of course those skilled in the art would appreciate the user of indicators to indicate users dragging options when in the event of dragging).

As for dependent claim 4, DragThing teaches the method as recited in claim 3, wherein the set of guides include a curved carat indicating the inclusion of a selected graphic control to a group and a straight line to indicate the exclusion of a selected graphic control from a group. However DragThing does not expressly disclose a carat icon or a straight-line icon for use of drag operations. It would have been obvious to one of ordinary skill in the art at the time of the invention to present any type of icon to represent a drag operation let alone differing ones to mark differing functions of drag operations, because DragThing discloses the user able to drag and organize as well as the user of icons for performing drag operations and of course, those skilled in the art would appreciate that a carat or straight-line graphic could be used to denote any function but more importantly differing functions as taught by DragThing (DT2, page 8, par.6 and DT2, page 16, par.4)

As for dependent claim 5, DragThing teaches the method as recited in claim 1, wherein displaying the first and the other than the first graphic controls as a group within the second portion of the display including displaying a graphic group control for instantiating an action on the first and the other than the first graphic controls and the respective graphical windows corresponding to the graphic controls (APPLE, page 2, par. 3; *resized, color coded, min and maximized*).

As for dependent claim 6, DragThing teaches the method as recited in claim 5, wherein instantiating an action on the first and the other than the first graphic controls is selected from a group consisting of minimizing the graphical windows corresponding to the first and the other than the first graphic controls (claim 5), restoring the graphical windows corresponding to the first and the other than the first graphic controls (APPLE, page 2, par.3 wherein the user can maximize or what is called in the art as window restoration ("to restore a window") as commonly defined in windows operating system help manuals), closing the graphical windows corresponding to the first and the other than the first graphic controls (APPLE, page 2, par.4; quit all running applications under a dock), saving data within the graphical windows corresponding to the first and the other than the first graphic controls (of course, those skilled in the art will appreciate that saving information within an application while its in view/ active to the user is commonly known and well understood), and resizing the graphical windows corresponding to the first and the other than the first graphic controls (claim 5).

As for dependent claim 7, DragThing teaches the method as recited in claim 1, wherein the displaying the first and the other than the first graphic controls as a group within the second portion of the display includes displaying at least a portion of the first and the other than the first graphic controls, the method further comprising: obtaining an indication to collapse the group; and displaying the group solely as a group graphic control (note the analysis of claims 1 and 5; putting the group into a tab, using minimize and maximize actions onto the tab graphic).

As for dependent claim 8, DragThing teaches the method as recited in claim 7, wherein obtaining an indication to collapse the group includes: monitoring the frequency of manipulation of the first and the other than the first graphic controls; and automatically collapsing the group if the frequency of manipulation is below a threshold level (APPLE, page 2,par.3; wherein the user can assign the action of minimize automatically which works as a delay when the user is not using the group, thus of course those skilled in the art would appreciate the use of a timer to complete such a task of timed interval that are exceeded from inactivity from the user to perform a function x).

As for dependent claim 9, DragThing teaches the method as recited in claim 7, wherein obtaining an indication to collapse the group includes obtaining a user indication to collapse the group (APPLE, page 2, par.3; wherein groups can be minimized, thus the user whom is operating the system has the option to collapse the group at anytime during interaction).

As for dependent claim 10, DragThing teaches the method as recited in claim 1, wherein rendering the first and the other than first graphic controls as a group within the second portion of the display includes associated a name with the group and renderings the name on a portion of the group indicator (DT2, pages, 17-19; labels for docks and items are using to indicate a plurality of items that differ from one another).

As for dependent claim 11, DragThing teaches the method as recited in claim 1, further comprising: obtaining an indication to remove the other than the first graphic control from the group; and rendering the first and the other than the first graphic controls as separate graphic controls (DT2, page 6).

As for dependent claim 12, DragThing teaches the method as recited in claim 1, wherein displaying the first and the other than the first graphic controls as a group within the second portion of the display includes displaying a continuous border around the first and the other than the first graphic controls indicating the association of the first and the other than the first graphic controls to the group (APPLE, page 2, figure; wherein as depicted in the figure the group called ‘application’ is color coded and has a border marking the interaction region that the group ‘application’ covers which controls are : finder, system pre…, acrobat, iTunes, etc.).

As for dependent claim 15, DragThing teaches the method as recited in claim 1, wherein the group corresponds to a project and wherein displaying the first and the other than the first graphic controls as a group within the second portion of the display includes displaying a project control for toggling the software applications corresponding to the first and the other than the first graphic controls between a minimized state and a restored state (note the analysis of claims 6,8,9).

As for dependent claim 16, the method as recited in claim 15, wherein rendering a project control includes rendering a project name and a number of software application include with the project (DT2, page 6).

As for dependent claim 17, DragThing teaches the method as recited in claim 15, wherein obtaining an indication to organize the first and the other than the first graphic controls includes obtaining a user manipulation of a selection device to drag and drop the first graphic control on the other than the first graphic control to organize the first and the other than the first graphic controls as a project (note the analysis of claim 2).

As for dependent claim 18, DragThing teaches the method as recited in claim 15, wherein the computer system includes a base project group separate from the project including the first and the other than the first graphic controls and wherein the one or more software applications belong to the base project (APPLE, page 2, par.5; wherein the user puts all of a certain project files under one group- these controls as which would be separate from the creation of the project files one could arrange an organization structure as noted in paragraph 5).

As for independent claim 48, DragThing teaches a computer system including a display comprising: plurality of software applications, each software application displaying plurality corresponding graphical windows and displaying one or more corresponding control tiles; a desktop for displaying the one or more graphical windows corresponding to each software application on the display; a taskbar for displaying the plurality control tiles corresponding to each software application on the display; a organization component for organizing the plurality control tiles in the taskbar, the organizing includes while the plurality of application are instantiated obtaining an indication to organize, selecting the plurality control tiles, displaying a set of guides for inclusion or exclusion into one or more group, releasing the plurality control tiles according to the indication to organize; a rendering component for displaying the plurality control tiles according to the indication to organize, if the indication to organize was for inclusion in a group, the rendering component displays the one or more control tiles within a group while the plurality of application are instantiated, the group including the plurality control tiles and a group control tile visually separated from other items in the taskbar.   (APPLE, page 2, paragraph 2); (note the analysis of claim 1; wherein the dock represents a user defined toolbars and the tabs represent user defined groups and the buttons/ icons representing application, files, etc represent tiles) (DT2, page 21, par.4 and page 18, par.2; page 12 and 6, wherein the user can group controls together in layers with group controls called tabs which are customizable by arrangement, color, font, name etc. In addition the user is presented with sorting options which will sort the application/icons by predefined conditions set in the options menu as defined by factory or user. Common to what is known in the art in the field of computational sorting is an automatic arrangement of data element defined by the rules of the sorting command currently executed; support for this reasoning can be found here: http://en.wikipedia.org/wiki/Sorting_algorithm). *In DT2 provides evidence that suggests that there is an indication of running processes (applications concurrently running on the system), these processes are dock items, that dock items can be located in a plurality of docks and that multiple dock items can be selected and dragged to new locations.*

DragThing does not specifically in detail explain that the system is capable of "preserving a layout, as rendered on the first portion of the display, of each graphical window that corresponds to each graphic control in the group”. However in the same field of endeavor **Westerman teaches** wherein a layout, as rendered on the desktop, of each graphical window corresponding to each control tile in the group is preserved (col.7, lines 40-46 to put this citation in better perspective In col.5, lines 5-32, 41-53; Westerman explains the grouping of windows and their presentation to the user). *It would have been obvious* to one of ordinary skill in the art at the time of the invention to combine Westerman into DragThing, this is true because Westerman teaches of solving the problem of organizing windows in to groups with the method of dismissing and calling groups of windows using the interface provided (col.2, lines 25-63). The combination of Westerman into DragThing yields the variant option of preserving the content layout for an interface which manages grouped content (i.e. windows).

As for dependent claim 49, DragThing teaches the method as recited in claim 48, wherein obtaining an indication to organize the one or more control tiles includes obtaining a user manipulation of a selection device to drag and drop the first control tile adjacent to the second control tile (note the analysis of claim 2).

As for dependent claim 50, DragThing teaches the method as recited in claim 48 further comprising displaying a set of guides indicating one or more possible groupings of control tiles corresponding to a drag and drop on a selected guide (note the analysis of claim 3).

As for dependent claim 51, DragThing teaches the method as recited in claim 50, wherein the set of guides include a curved carat indicating the inclusion of a selected control tile to a group and a straight line to indicate the exclusion of a selected control tile from a group (note the analysis of claim 4).

As for dependent claim 52, DragThing teaches the method as recited in claim 48, wherein the group control tile includes one or more group controls (note the analysis of claim 5).

As for dependent claim 53, DragThing teaches the method as recited in claim 52 wherein instantiating an action on the plurality control tiles is selected from a group consisting of minimizing the graphical windows corresponding to the plurality control tiles, restoring the graphical windows corresponding to the plurality control tiles, closing the graphical windows corresponding to the plurality control tiles, saving data within the graphical windows corresponding to the plurality control tiles, and resizing the graphical windows corresponding to the plurality control tiles (note the analysis of claim 6).

As for dependent claim 54, DragThing teaches the method as recited in claim 48, wherein the displaying the plurality control tiles as a group within the taskbar portion of the display includes displaying at least a portion of the plurality control tiles, the method further comprising: obtaining an indication to collapse the group; and displaying the group solely as a group control tile (note the analysis of claim 7).

As for dependent claim 55, DragThing teaches the system as recited in claim 48, wherein displaying the plurality control tiles as [[a]]the group within the taskbar portion of the display includes associating a name with the group and displaying the name on a portion of a group indicator (DT2. page 17-19).

As for dependent claim 56, DragThing teaches the system as recited in claim 48 further comprising: the organization component obtaining an indication to remove plurality tile from the group; and the rendering component displaying the plurality control tiles removed from the group as separate control tiles (DT2, page 6-7).

As for dependent claim 57, DragThing teaches the method as recited in claim 48, wherein displaying the plurality control tiles as a group within the taskbar portion of the display includes displaying a continuous border around the plurality control tiles indicating the association of the plurality control tiles to the group (note the analysis of claim 12).

As for dependent claim 58, DragThing teaches the method as recited in claim 57, wherein displaying a continuous border around the plurality control tiles indicating the association of the plurality control tiles to the group includes displaying the continuous border in a color separate from a color corresponding to the taskbar (APPLE, page 2, par.2; color coded).

As for dependent claim 59, DragThing teaches the method as recited in claim 58 wherein displaying a continuous border around the plurality control tiles indicating the association of the plurality control tiles to the group includes displaying the continuous border in a color separate from any other color of a group on the taskbar (APPLE, page 2, par.2; border is colored black as depicted in the figure which is separate from the yellow, red, green, etc group colors).

As for dependent claim 60, DragThing teaches the method as recited in claim 48, wherein the computer system includes a third software application represented as graphical windows in the desktop portion of the display and as a control tile on the taskbar portion of the display when the third software application is instantiated on the computer system, the method further comprising: obtaining an indication to organize a control tile corresponding to the third software application into the group corresponding to the plurality control tile; grouping the third control tile with the plurality control tiles on the taskbar portion of the display; and displaying the first, second and third control tiles as a group within the taskbar portion of the display (note the analysis of claim 13).

As for dependent claim 61, DragThing teaches the method as recited in claim 48, wherein the group corresponds to a project and wherein displaying the plurality control tiles as a group within the taskbar portion of the display includes displaying a project control for toggling the software applications corresponding to the plurality control tiles between a minimized state and a restored state (note the analysis of claim 15).

As for dependent claim 63, DragThing teaches the method as recited in claim 61, wherein the computer system includes a base project separate from the project including the plurality control tiles and wherein the plurality software applications belong to the base project (note the analysis of claim 18).

As for independent claim 84, DragThing teaches a computer system for managing a plurality of software applications, the system comprising: means for displaying two or more software applications as graphical windows when the two or more software applications are instantiated on the computer system; means for displaying two or more control tiles corresponding two or more software applications when the two or more software applications are instantiated on the computer system; and means for displaying the two or more control tiles as group in response to indication to organize the one or more control tiles (note the analysis of claim 13) and means for automatically, displaying the two or more control tiles in a group based on an action to organize control tiles, the action to organize triggers a set of guides indicating inclusion or exclusion of a group and places the tiles within or without the group relative to the set of guides, the group includes the two or more control a group within the second portion of the display, the group control tile configured for simultaneous action on both the first instantiated software application and the second instantiated software application (DT2, page 21, par.4 and page 18, par.2; page 12 and 6, wherein the user can group controls together in layers with group controls called tabs which are customizable by arrangement, color, font, name etc. In addition the user is presented with sorting options which will sort the application/icons by predefined conditions set in the options menu as defined by factory or user. Common to what is known in the art in the field of computational sorting is an automatic arrangement of data element defined by the rules of the sorting command currently executed; support for this reasoning can be found here: http://en.wikipedia.org/wiki/Sorting_algorithm).

DragThing does not specifically in detail explain that the system is capable of "preserving a layout, as rendered on the first portion of the display, of each graphical window that corresponds to each graphic control in the group". However in the same field of endeavor **Westerman teaches** wherein a layout, as rendered when the software applications are displayed, each graphical window corresponding to each control tile in the group is preserved (col.7, lines 40-46 to put this citation in better perspective In col.5, lines 5-32, 41-53; Westerman explains the grouping of windows and their presentation to the user). *It would have been obvious* to one of ordinary skill in the art at the time of the invention to combine Westerman into DragThing, this is true because Westerman teaches of solving the problem of organizing windows in to groups with the method of dismissing and calling groups of windows using the interface provided (col.2, lines 25-63). The combination of Westerman into DragThing yields the variant option of preserving the content layout for an interface which manages grouped content (i.e. windows).

As for dependent claim 85, DragThing teaches the system as recited in claim 84, wherein the means for displaying the two or more control tiles includes means for generating a set of guides indicating one or more possible organizations of control tiles (note the analysis of claim 3).

As for dependent claim 86, DragThing teaches the system as recited in claim 84 the simultaneous action on both the first instantiated software application and the second instantiated software application includes: minimizing the graphical window corresponding to each software application, restoring the graphical window corresponding to each software application, closing the graphical window corresponding to each software application, saving data within the graphical window corresponding to each software application, or resizing the graphical windows corresponding to each software application (DT2, page 2, 6-7).

As for dependent claim 87, DragThing teaches the system as recited in claim 84, wherein the means for displaying the two or more control tiles as a group includes means for displaying a group control without displaying any portion of the two or more control tiles (note the analysis of claim 7).

As for dependent claim 93, DragThing teaches the system as recited in claim 84 further comprising: means for obtaining an indication to collapse the two or more control tiles into the group control tile; and means for displaying the group solely as a group control tile (DT2, page 6, 16-18).

As for dependent claim 94, DragThing teaches the system as recited in claim 84 further comprising: means for obtaining an indication to remove two or more control tiles from the group; and means for displaying the two or more control tiles removed from the group as separate control tiles (DT2, page 6, 16-18).

(Note :) **It is noted that any citation to specific, pages, columns, lines, or figures in the prior art references and any interpretation of the references should not be considered to be limiting in any way. A reference is relevant for all it contains and may be relied upon for all that it would have reasonably suggested to one having ordinary skill in the art. In re Heck, 699 F.2d 1331, 1332-33, 216 USPQ 1038, 1039 (Fed. Cir. 1983) (quoting In re Lemelson, 397 F.2d 1006,1009, 158 USPQ 275, 277 (CCPA 1968)).**

Response to Arguments

Applicant's arguments filed 11/04/2009 have been fully considered but they are not persuasive.

A1. Applicant argues that DragThing does not teach preserving windows layout.

R1. Examiner agrees, DragThing does not clearly detail that the windows layout along with position is preserved such that when the user clicks or activates the group again that the windows positions were preserved. However Westerman is introduced to clear the deficiencies of DragThing by adding a variant option of preserving windows layout.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Inquires

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nicholas Augustine whose telephone number is 571-270-1056 and fax is 571-270-2056. The examiner can normally be reached on Monday - Friday: 9:30am- 5:00pm Eastern.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Weilun Lo can be reached on 571-272-4847. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Nicholas Augustine/
Examiner
Art Unit 2179
January 15, 2010

/Ba Huynh/
Primary Examiner, Art Unit 2179